

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A method of storing data pages at a proxy, the method comprising:

a) receiving a data page from an origin server, wherein said data page includes page dependency data that contains one or more dependencies such that each dependency indicates an underlying data source which said data page is dependent on, wherein said page dependency data are generated by a Request-Based dependency generator which uses a configuration file corresponding to the origin server, wherein the configuration file is the only configuration file corresponding to the origin server;

b) storing said data page;

c) storing said page dependency data;

d) receiving an event;

e) determining if said event changes an underlying data source corresponding to one of said page dependency data; and

f) updating the cache by refreshing or deleting said data page if said event changes one of the page dependency data.

2. (Original) A method as defined in claim 1, where said page dependency data are written in HTML or XML.

3. (Currently Amended) A method as defined in claim 1, wherein said page dependency data are listed between specially delimited comments appended to the end of said data page ~~where said page dependency data are generated by a Request-Based dependency generator.~~

4. (Original) A method as defined in claim 3, where said Request-Based dependency generator uses a URL request of said data page.

5. (Canceled)

6. (Original) A method as defined in claim 1, where said page dependency data are generated by a script-based dependency generator.

7. (Original) A method as defined in claim 6, where said page dependency data are encoded in said data page by a script-based dependency generator.

8. (Original) A method as defined in claim 1, where said page dependency data are manually encoded into a data file.

9 (Original) A method as defined in claim 1, where said data page and said page dependency data are stored in one or more files.

10. (Original) A method as defined in claim 1, where said storing of said data page at the proxy is in response to data in a configuration file.

11. (Canceled)
12. (Previously Presented) A method as defined in claim 1, where send event is received incorporated in an event message.
13. (Previously Presented) A method as defined in claim 1, where said event is written in HTML or XML.
14. (Previously Presented) A method as defined in claim 1, where said event came from a Request-Based event generator.
15. (Original) A method as defined in claim 14, where said Request-Based event generator uses a configuration file.
16. (Original) A method as defined in claim 14, where said Request-Based event generator uses a URL request of said data page.
17. (Original) A method as defined in claim 16, where said URL request is parsed to obtain parameters.
18. (Original) A method as defined in claim 16, where said tJRL request includes request header information.

19. (Original) A method as defined in claim 18, where said tJRL request is parsed to obtain parameters.
20. (Previously Presented) A method as defined in claim 1, where said event came from a script-based event generator.
21. (Previously Presented) A method as defined in claim 1, where said event came from a trigger-based event generator.
22. (Previously Presented) A method as defined in claim 1, where said event came from polling event generator.
23. (Previously Presented) A method as defined in claim 1, where said event came from a custom event generator.
24. (Previously Presented) A method as defined in claim 1, where said determination if said event changes one of the page dependency data associated with said data page is done by a change event evaluator.
25. (Previously Presented) A method as defined in claim 1, where said determination by the change event evaluator is done by matching said page dependency data with said event.

26. (Previously Presented) A method as defined in claim 1, where said event was generated by Request-Based event generator and sent to a change event evaluator.
27. (Previously Presented) A method as defined in claim 1, where said updating the cache involves keeping the index of URL addresses and page dependency data up-to-date.
28. (Currently Amended) A computer software product for use in a computer system that executes program steps recorded in a computer-readable media to perform a method for caching dynamic data pages enabling storing of data pages at a proxy comprising:
- a) a recordable media; and
  - b) a program of computer-readable instructions executable by the computer to perform method steps comprising:
    - i) receiving a data page from an origin server, wherein said data page includes page dependency data that contains one or more dependencies such that each dependency indicates an underlying data source which the said data page is dependent on, wherein said page dependency data are generated by a Request-Based dependency generator which uses a configuration file corresponding to the origin server, wherein the configuration file is the only configuration file corresponding to the origin server;
    - ii) storing said data page;
    - iii) storing said page dependency data;
    - iv) receiving an event;

- v) determining if said event changes said one of the page dependency data associated with said data page; and
- vi) updating the cache by refreshing or deleting said data page if said event changes said one of the page dependency data associated with said data page.

29. (Currently Amended) A proxy server system that provides stored data files without requesting data files from the origin web server, comprising:

- a) a central processing unit that can establish communication with a user computer;
- b) a storage device;
- c) a processor connected to the storage device wherein the storage device stores:
- d) the processor is operative with said program component to:
  - i) receive a data page from an origin server, wherein said data page includes page dependency data that contains one or more dependencies such that each dependency indicates an underlying data source which the said data page is dependent on, wherein said page dependency data are generated by a Request-Based dependency generator which uses a configuration file corresponding to the origin server, wherein the configuration file is the only configuration file corresponding to the origin server;
  - ii) store said data page;
  - iii) store said page dependency data;
  - iv) receive an event;

- v) determine if said event changes said one of the page dependency data associated with said data page; and
- vi) update the cache by refreshing or deleting said data page if said event changes said one of the page dependency data associated with said data page.

30-36. (Canceled)

37 (Newly Added) A method for caching a dynamic web page, the method comprising:

receiving a request for the dynamic web page at an origin server, the request sent from a dynamic content cache;

reading a configuration file created for the origin server, the configuration file containing one or more rules to be applied to dynamic web pages relating to the origin server in order to generate dependencies, wherein the configuration file is the only configuration file for the origin server;

generating the dynamic web page;

generating one or more dependencies for the dynamic web page by applying the one or more rules to the dynamic web page;

associating the one or more dependencies with the dynamic web page; and

sending the dynamic web page and the associated dependencies to the dynamic content cache for storage.

38 (Newly Added) The method of claim 37, wherein said one or more rules include URL pattern matching. (Page 27, line 15-22)

39. (Newly Added) The method of claim 37, wherein said associating includes appending the one or more dependencies at the end of the dynamic web page. (Page 26, lines 16-18\_

40. (Newly Added) The method of claim 37, wherein the one or more dependencies for the dynamic web page do not include dependencies for at least one object which is to be displayed when the dynamic web page is loaded by a browser.

41 (Newly Added) The method of claim 40, wherein said dependencies for at least one object which are not included in the one or more dependencies for the dynamic web page are determined by one or more rules indicating non-cacheable objects, the rules included in the configuration file.

42. (Newly Added) The method of claim 37, wherein the one or more dependencies are written in XML